**Loan Default Prediction**

*A project report submitted to ICT Academy of Kerala*

*in partial fulfillment of the requirements*

*for the certification of*

**CERTIFIED SPECIALIST**

**IN**

**DATA SCIENCE & ANALYTICS**

submitted by

**Aishwarya N**

**Anjitha Ashok A**

**Linu Paulose**

**Thamanna Parveen**



**ICT ACADEMY OF KERALA**

**THIRUVANANTHAPURAM, KERALA, INDIA**

**Nov 2022**

**List of Abbreviations**

* FP-False Positive
* FN-False Negative
* TP-True Positive
* TN-True Negative
* LR-Logistic Regression
* KNN-K-nearest neighbor algorithm
* XG-BOoster-Extreme Gradient Boosting
* SVM-Support Vector Machine
* NN-Neural Network

**Table of Contents**

**Abstract**  **5**

1. **Problem Definition**  6

* 1.1 Overview
* 1.2 Problem Statement

1. **Introduction 7**
2. **Literature Survey 8**
3. **Dataset and Characteristics 9**
4. **Data Preprocessing**  11

* 5.1 Handling Missing Values 13-33
* 5.2 Outlier Detection
* 5.3 FeatureEncoding
* 5.4 Feature Scaling
* 5.5 Feature Construction
* 5.6 Feature Selection
* 5.7 Dimensionality Reduction

1. **Model Building 34-42**

* 6.1 LOGISTIC REGRESSION
* 6.2 EXTREME-GRADIENT BOOSTING
* 6.3 RANDOM FOREST
* 6.4 DECISION TREE
* 6.5 SVM
* 6.6 KNN

1. **Results 48**
2. **Conclusion 50**
3. **Reference 53**

**Abstract**

Since there are so many people taking out loans,the chances of the loans being defaulted is also high. so it is very necessary for banks to identify which type of customers have most probability for a lan default.If this process of identification is carried out manually, it might be even more difficult.As a solution for this problem we decided to create a machine learning-based loan prediction system that will predict with their past action that if there is a possibility of loan default or not.Both the applicant and the bank staff will benefit from this. To build a model for predicting the probability of default of the applicant we used the loan default dataset from kaggle.In the data we used machine learning classification algorithms to identify the right clients to target for loan default.We tried to evaluate and visualize all the features that may influence and causes the loan default.

**1. Problem Definition**

**1.1 Overview**

Loan Default is the failure to make required interest or payments on a debt,as per the terms agreed between customer and lender.Extended delinquency can result in loan default.Default can occur on secured loan such as a mortgage loan secured by a house,unsecured loan such as credit card.

**1.2 Problem Statement**

The Loan Default Dataset is to classify borrower as defaulter or non-defaulter.Financial Institutions uses credit scoring models to evaluate potential loan default risk.Developing credit scoring model is time consuming.Credit scoring models are fixed and do not easily evolve with changing customer behavior to predict the default more accurately,these procedures are inefficient and time consuming.In order to overcome this we use Machine Learning model.

Machine Learning approaches can help enhance the accuracy of loan default prediction.ML is more dynamic and adaptable to changing client data.Hence the idea of the project is to gather the loan default data from dataset and use machine learning techniques on this data to extract important information and predict if a customer would be able to repay or not.The goal is to predict if the customer would be a defaulter or non-defaulter.

**2. Introduction**

Individuals all around the world in some way depend on banks to lend them loans for various reasons to help them overcome their financial constraints and achieve some personal goals. Due to the ever changing economy and ever-increasing competition in the financial world, the activity of taking a loan has become inevitable. A loan is the major source of income for the banking sector as well as the biggest source of financial risk for banks. In today’s world, obtaining loans from financial institutions has become a very common phenomenon. Every day many people apply for loans, for a variety of purposes. Every year, there are cases where people do not repay the bulk of the loan amount to the bank which resulting huge financial losses. A loan default occurs when a borrower takes money from a bank and does not repay the loan. People often default on loans due to various reasons. Borrowers who default on loans not only damage their credit but also risk being sued and having their wages garnished.

Hence, the idea of this project is to analyze the chosen loan data from Kaggle and use machine learning techniques on this data to extract important information and predict if a customer would be able to repay the loan or not. In other words, the goal is to predict if the customer would be a defaulter or not. The provided dataset corresponds to loans issued to individuals in the year 2019 in the context of India. The dataset has 148670 observations and 34 features. The data contains loan amount, interest rate, gender, loan purpose, payment information, credit score, debt-to-income ratio, loan-to-value, and so on. The status column shows the current state of each loan record. We used classifiers to analyze the dataset and identify the patterns in the dataset.

**3. Literature Survey**

In the research paper titled “ Loan Default Prediction with Machine Learning”by Tomi Himberg under the supervision of Dr. Xiaolu Wang in 2021 they have applied a number of supervised techniques to predict the default status of loans.The algorithms used in the study were logistic regression, classification tree, random forest and extreme gradient boosting.The logistic regression model predicted 63.3% of the defaulted loans correctly and 62.1% of the non-defaulted loans in their study.The random forest model predicts non defaulted loans better than defaulted loans with an accuracy of 64.1%.An oversampling and undersampling were used in the model creation in Classification trees.The undersampled data and oversampled data had accuracies of 53.2% and 61.8% respectively.The random forest model predicts non defaulted loans better than defaulted loans with an accuracy of 64.1% The XGBoost identifies 70% of defaulted loans correctly on unseen data.Based on these results, the XGBoost model obtained the most precise and powerful results on loan default prediction.

In the project on “Loan default prediction” submitted by Vikas. V and Mohammad Amir Ahmed in 2018 have found logistic regression to be the best model in predicting the loan default with an accuracy of 99% and a precision of 40%.And Random Forest was found to have 78% of accuracy.The model was able to predict who are going to pay off the loan with a good accuracy of 99% but was unable to predict who were going to default

Another research paper titled “Predicting Educational Loan Defaults: Application of Machine Learning and Deep Learning Models” by Jayadev M, Professor ,IIM,Bangalore has performed a number of models in the data like Logistic regression,Decision Trees,SVM,random Forest and KNN. For most models, precision exceeded 75% with tree-based ensembles having the highest among the group.

**4. Dataset and Its Characteristics**

We used the dataset [“LOAN-DEFAULT PREDICTION](https://docs.google.com/spreadsheets/u/0/d/14di-RqyElWQYQF241nHKqrSH3PkcdBr7/edit)” from Kaggle.It has 10,000 rows and 28 columns,which makes it an ideal data set for applying a ML algorithm.It has both Categorical and Numerical Columns.

**Categorical Variables**

* Loan limit
* Gender
* Approve in advance
* Loan type
* Loan purpose
* Creditworthiness
* Open credit
* Business or commercial
* Neg amortization
* Interest only
* Lump sum payment
* Construction type
* Occupancy type
* Gender
* Secured by
* Total units
* Credit type
* Co-applicant Credit type
* Age
* Submission of application
* Region
* Security type
* Status

**Numerical Variables**

* Unnamed
* ID
* Year
* Loan amount
* Rate of interest
* Interest rate spread
* Upfront charges
* Term
* Property value
* Income
* Credit score
* LTV
* Debt to Income Ratio

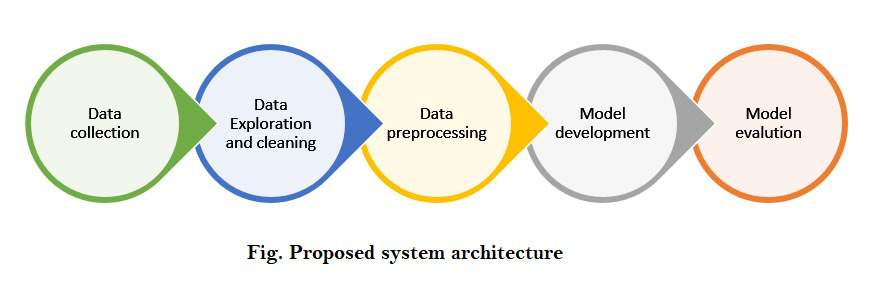
By going through the data,since the columns titled “ID” and “Year”(the year was the same for all the customers) seemed irrelevant in deciding our target column we decided to remove it.

**5.** **DATA PREPROCESSING**

Data preprocessing is the method of analyzing, filtering, transforming and encoding data so that a machine learning algorithm can understand and work with the processed output.

As the popular saying goes, “if garbage goes in, garbage comes out”. The data project can only be successful if the data going into the machines is high quality.A real-world data generally contains noises, missing values, and maybe in an unusable format which cannot be directly used for machine learning models. Data preprocessing is required for cleaning the data and making it suitable for a machine learning model which also increases the accuracy and efficiency of a machine learning model.

**FLOW DIAGRAM OF DATA SCIENCE PROCESS**

****

**Steps involved in data preprocessing:**

1. Acquire the dataset
2. Import all the crucial libraries
3. Import the dataset
4. Identifying and handling the missing values
5. Encoding the categorical data
6. Splitting the dataset
7. Feature scaling

**5.1 HANDLING MISSING VALUES:**

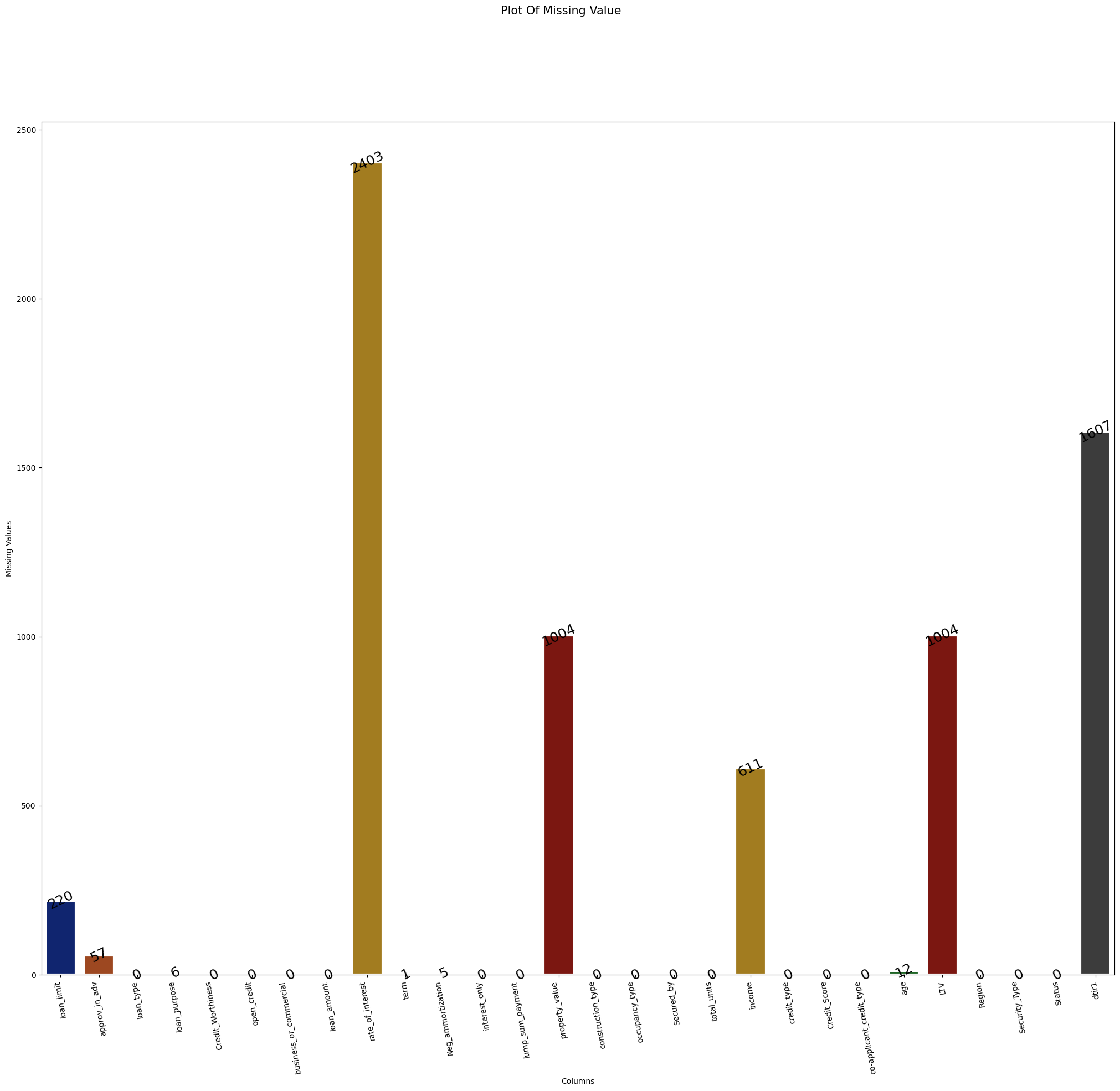
Missing data is defined as the values or data that is not stored (or not present) for some variable/s in the given dataset.In data preprocessing, it is pivotal to identify and correctly handle the missing values, failing to do this, you might draw inaccurate and faulty conclusions and inferences from the data.

**Methods to handle missing values:**

**Deleting the Missing values:** In this method, you remove a specific row that has a null value for a feature or a particular column where more than 75% of the values are missing. However, this method is not 100% efficient, and it is recommended that you use it only when the dataset has adequate samples. You must ensure that after deleting the data, there remains no addition of bias.

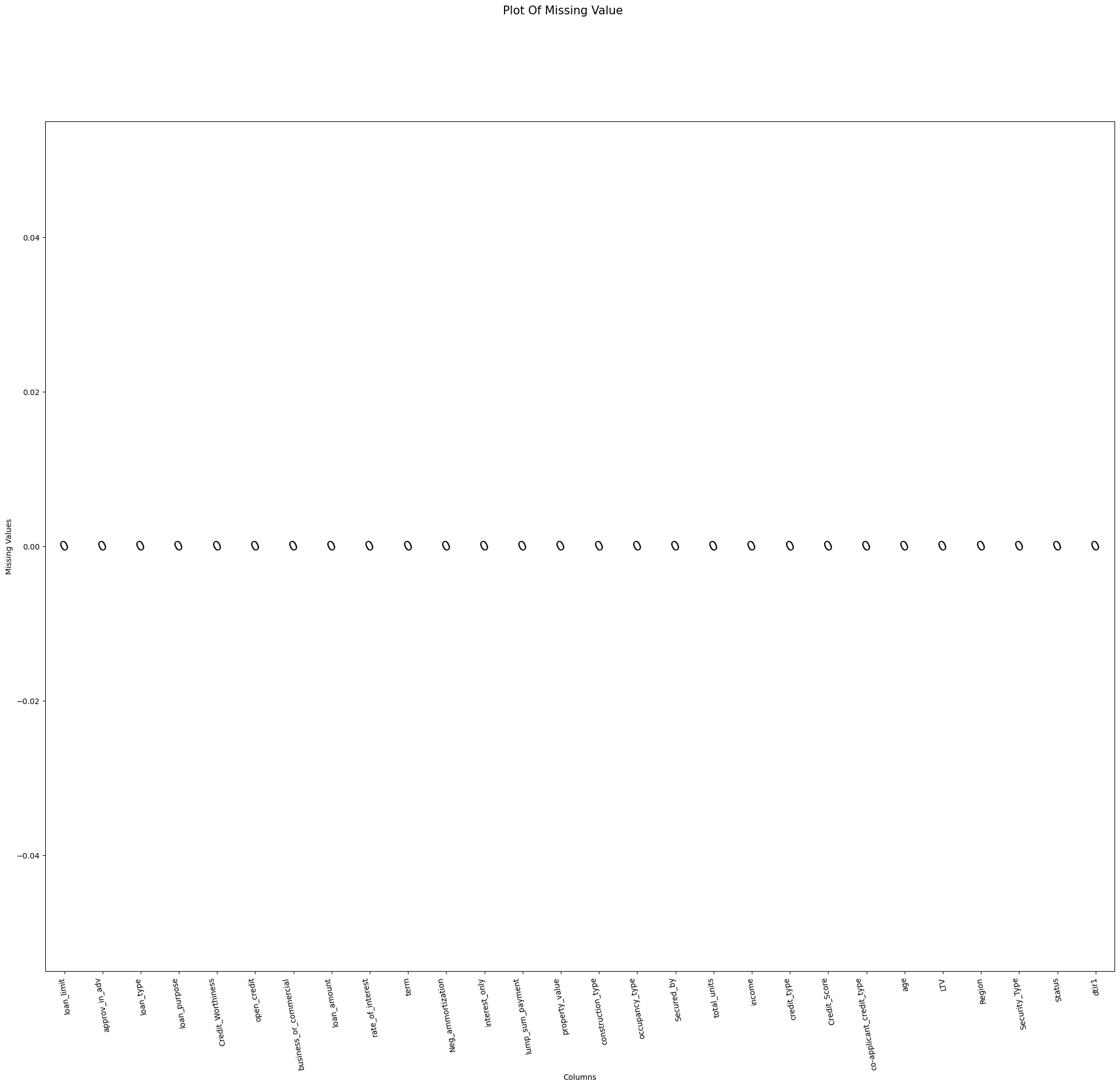
**Imputing the Missing Values:** This method is useful for features having numeric data like age, salary, year, etc. Here, you can calculate the mean, median, or mode of a particular feature or column or row that contains a missing value and replace the result for the missing value. This method can add variance to the dataset, and any loss of data can be efficiently negated. Hence, it yields better results compared to the first method (omission of rows/columns).

We checked for missing values in the data and a large number of missing values were found in most of the columns.The missing values in each column is represented as a bar chart below.

**PLOT OF MISSING VALUE** 

**fig 3.1**

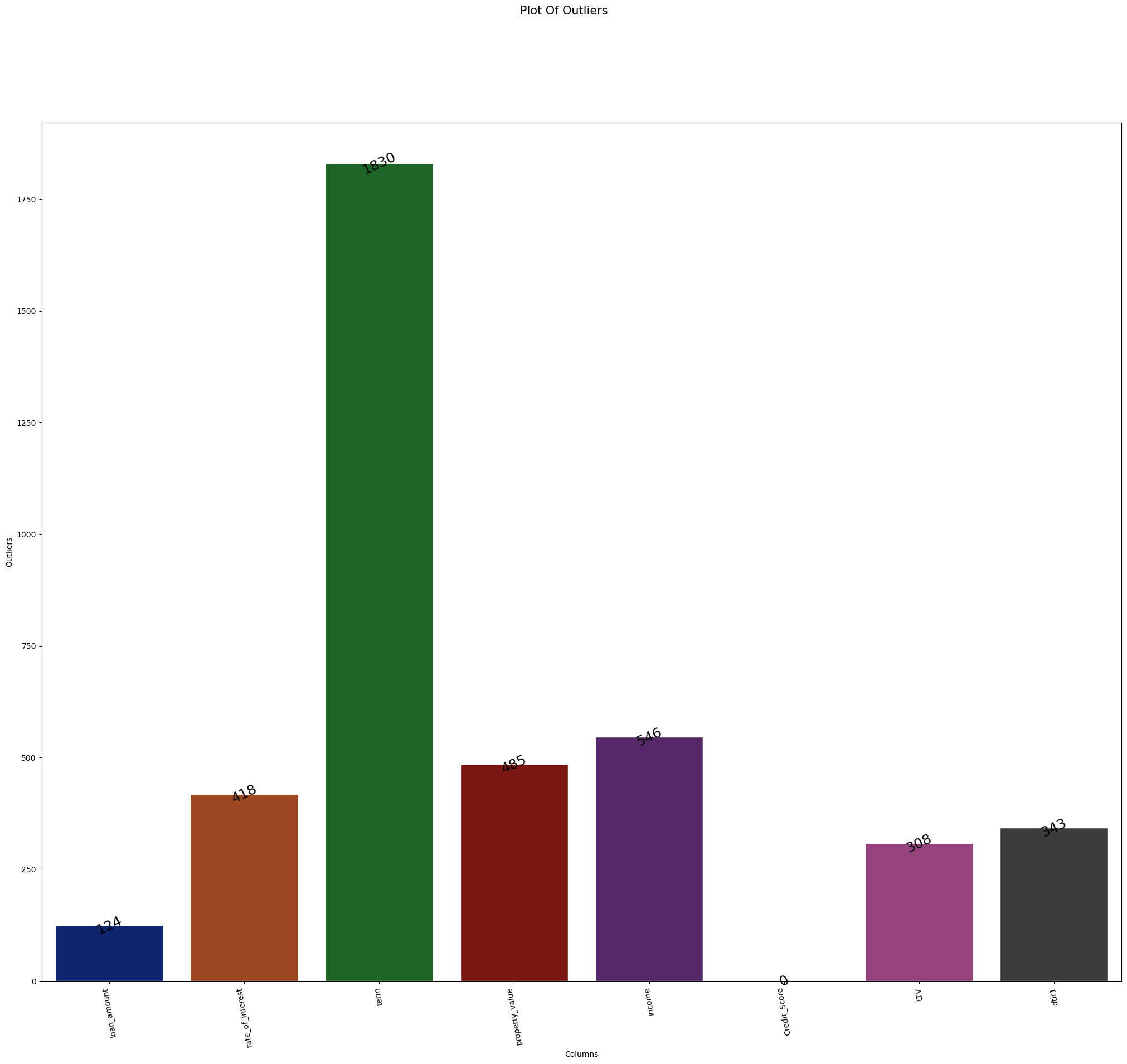
In order to treat the missing values,we used **Imputing the Missing Values Method** because, it preserves all cases by replacing the missing data with a probable value estimated by other available information.For LTV, Income, Property value ,data is numeric and not skewed,so we used the Mean to replace the missing values.Rate of Interest, Term, Debt to Income ratio,data is numeric and skewed,so we used the median to replace the missing value.Loan limit, Loan purpose, Approve in advance, Negative Amortization, Age,data is a string(object),so we used the mode to replace the missing values.The plot of the frequency of missing values after treatment is shown below.



**5.2 OUTLIER DETECTION**

Outliers are data points that do not conform with the predominant pattern observed in the data. They can cause disruptions in the predictions by taking the calculations off the actual pattern.Outliers can be detected and treated with the help of box-plots. Box plots are used to identify the median, interquartile ranges and outliers. To remove the outliers, the maximum and minimum range needs to be noted, and the variable can be filtered accordingly.We used IQR method to find the outliers.And then used an user-defined function to find the number of outliers using the lower limit and upper limit calculated using IQR method.

By drawing boxplots for each column and by using the IQR method we found out the frequency of outliers in each column and it is represented as a bar chart below.We have found the number of outliers for the numerical columns only.(since that only seemed logical).

 fig 3

Since outliers seemed so important for our prediction we kept the outliers in our data intact.

**5.3 FEATURE ENCODING:**

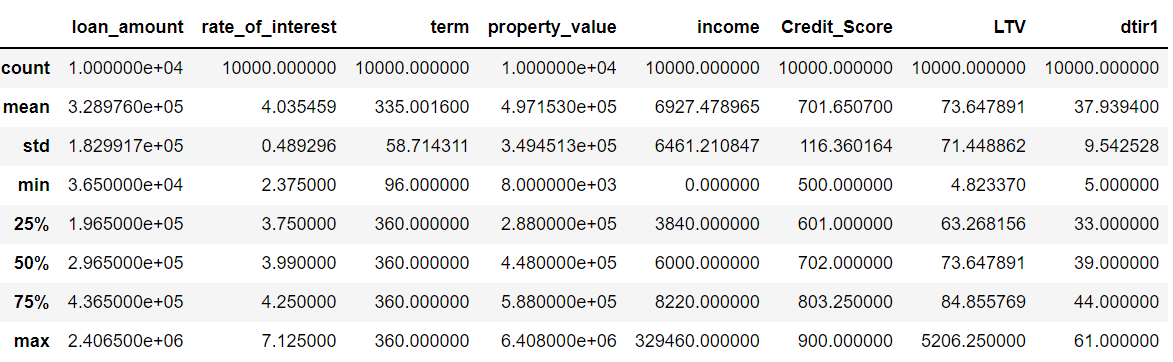
Encoding is a technique of converting categorical variables into numerical values so that it could be easily fitted to a machine learning model. There are mainly two types of encoding, label encoding and one hot encoding. In order to work with the categorical features in our data,we used one hot encoding,because the categorical features present in the data are not ordinal. After encoding the shape of the data became (10000,28) to (10000,64).

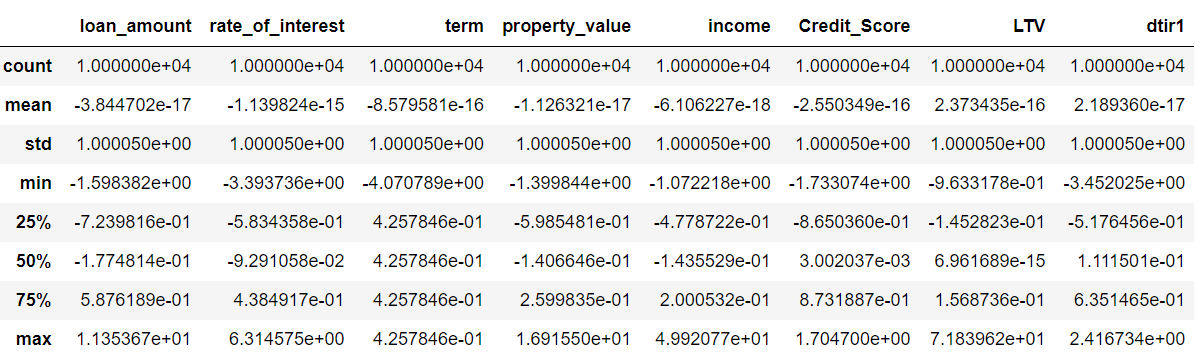
**5.4 FEATURE SCALING:**

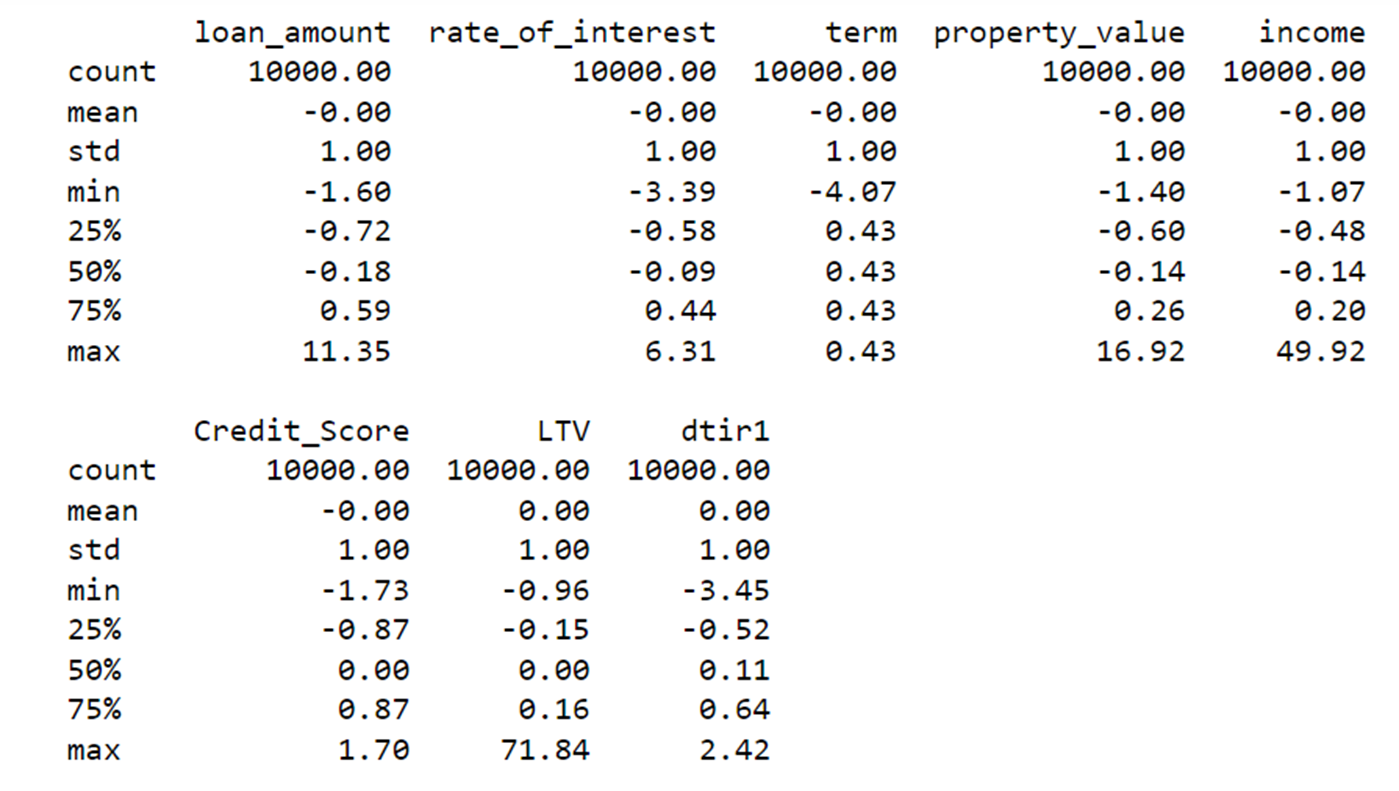
Feature Scaling is a technique to standardize the independent features present in the data in a fixed range.If feature scaling is not done, then a machine learning algorithm tends to weigh greater values, higher and consider smaller values as the lower values, regardless of the unit of the values. There are mainly 4 methods of scaling.

* Standard Scaler
* Min-Max Scaler
* Robust Scaler
* Max-Abs Scaler

**Description of the data**

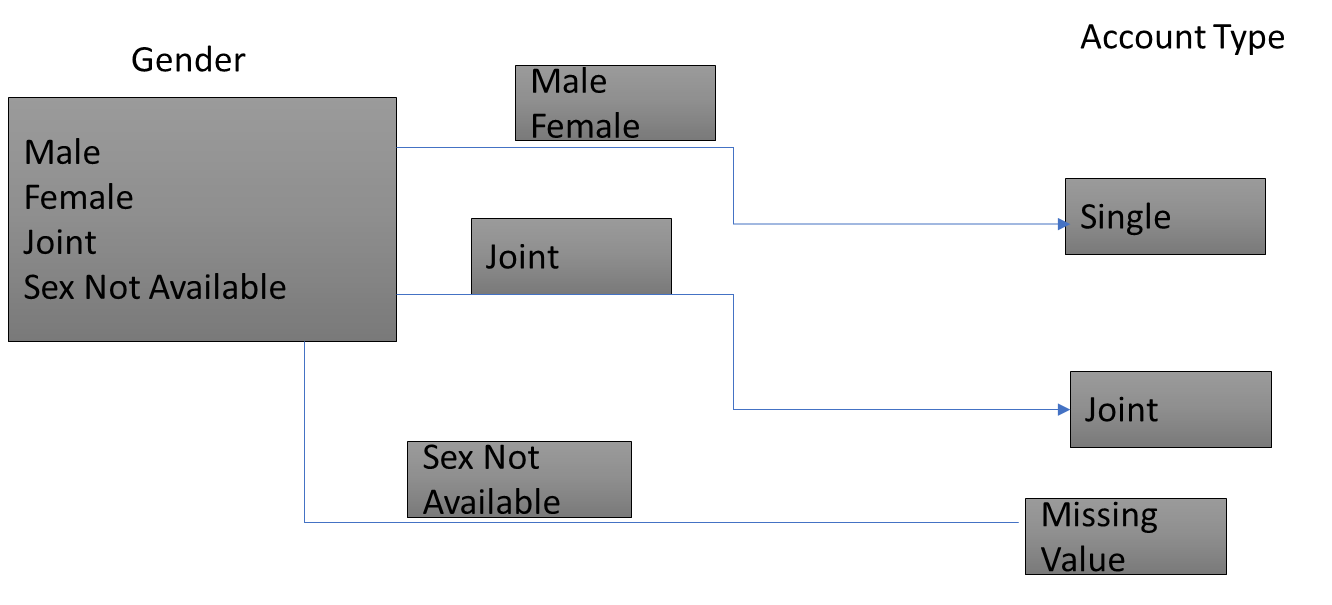


If you go through the description of the data,you could infer that the input dataset differs greatly between their ranges, and they are measured on different units of measure.The central tendency and dispersion measures of each features are entirely different to each other.We applied Standard Scaling in the data as a solution to this problem.Resulting data description is shown below.

****

**5.5 FEATURE CONSTRUCTION**

Higher-level features can be obtained from already available features and added to the feature vector; for example, for the study of diseases the feature 'Age' is useful and is defined as *Age = 'Year of death' minus 'Year of birth'* . This process is referred to as feature construction.Feature construction is the application of a set of constructive operators to a set of existing features resulting in construction of new features. Examples of such constructive operators include checking for the equality conditions {=, ≠}, the arithmetic operators {+,−,×, /}, the array operators {max(S), min(S), average(S)} as well as other more sophisticated operators, for example count(S,C) that counts the number of features in the feature vector S satisfying some condition C or, for example, distances to other recognition classes generalized by some accepting device. Feature construction has long been considered a powerful tool for increasing both accuracy and understanding of structure, particularly in high-dimensional problems.



In our data we had this column called Gender with categories Male,Female,Joint and Sex Not Available.From this column we constructed a column called Account Type having categories Single and Joint. Male and Female were converted to single account type and Joint was retained as joint.And Sex not available was treated as missing column and was replaced by the mode value of the account type.We think this column and its categories make much more sense.

**5.6 FEATURE SELECTION**

Feature selection is the process of reducing the number of input variables when developing a predictive model.

It is desirable to reduce the number of input variables to both reduce the computational cost of modeling and, in some cases, to improve the performance of the model.

**Feature Selection**: Select a subset of input features from the dataset.

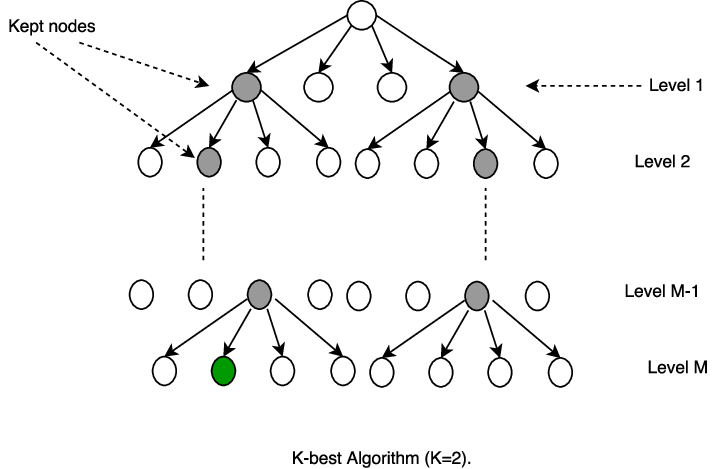
* **Unsupervised**: Do not use the target variable (e.g. remove redundant variables).
  + Correlation
* **Supervised**: Use the target variable (e.g. remove irrelevant variables)
  + - **Wrapper**: Search for well-performing subsets of features.
      * RFE
    - **Filter**: Select subsets of features based on their relationship with the target.
      * Statistical Methods
      * Feature Importance Methods
    - **Intrinsic**: Algorithms that perform automatic feature selection during training.
      * Decision Trees

In the feature selection method,we used the methods like

* **.**K-Best Algorithm
* Extra-Tree Classifier
* Information Gain
* Chi Square Test
* T test

**5.6.1 KBEST ALGORITHM**

Scikit-learn API provides SelectKBest class for extracting best features of given dataset. The SelectKBest method selects the features according to the k highest score. By changing the 'score\_func' parameter we can apply the method for both classification and regression data. Selecting the best features is an important process when we prepare a large dataset for training. It helps us to eliminate less important parts of the data and reduce training time.



# Visual Representation of K-BESTClassifier

| Features | Score |
| --- | --- |
| property\_value | 3.57E+06 |
| loan\_amount | 5.73E+05 |
| income | 2.28E+05 |
| dtir1 | 2.16E+02 |
| LTV | 1.97E+02 |
| co-applicant\_credit\_type | 1.35E+02 |
| credit\_type | 1.27E+02 |
| loan\_limit | 3.36E+01 |
| Region | 3.13E+01 |
| loan\_type | 3.02E+01 |

In our data,we used K-best Classifier to find the feature score of the independent features.The table below shows the first 10 important features that influence the target variable “Default Status” the most.Property Value ,Loan Amount etc…are found to have a higher influence on the target column than the others..Security Type and Secured by etc…had least scores based on the K-Best algorithm suggesting that they are least important in deciding the default status.

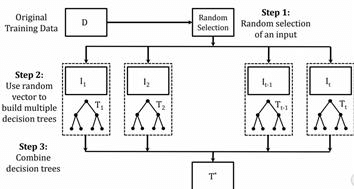
**5.6.2 EXTRA TREE CLASSIFIER**

The extra trees algorithm, like the random forests algorithm, creates many decision trees, but the sampling for each tree is random, without replacement. This creates a dataset for each tree with unique samples. A specific number of features, from the total set of features, are also selected randomly for each tree.

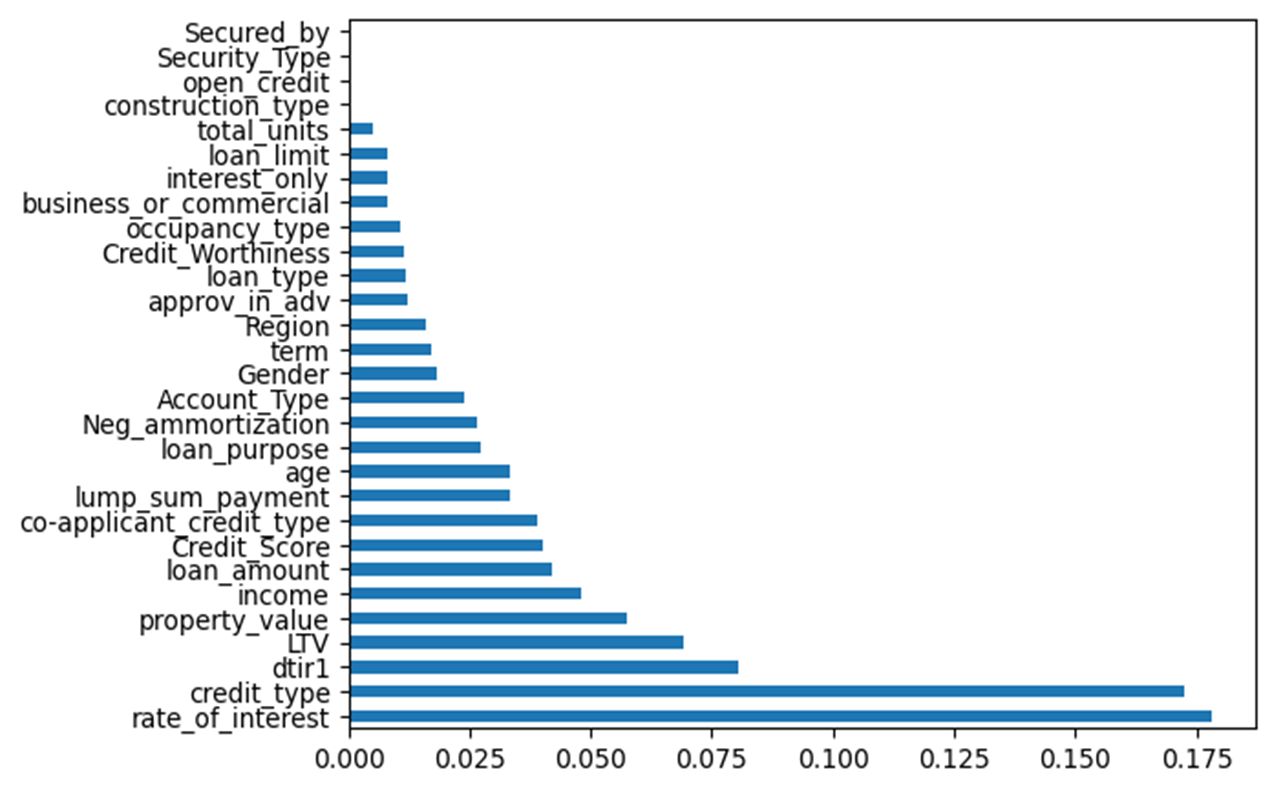
The most important and unique characteristic of extra trees is the random selection of a splitting value for a feature. Instead of calculating a locally optimal value using Gini or entropy to split the data, the algorithm randomly selects a split value. This makes the trees diversified and uncorrelated.

# 

# Visual Representation of Extra Trees Classifier



We used Extra tree Classifier also to find out the feature relevance.And found “Rate of interest” and “Credit type” to have a higher feature score.That is the change in them influences the loan default status the most.From the bar graph it is very evident that the features “Secured\_by”, “Security\_Type”, “Construction\_type”etc..have little to no relevance in deciding the target column.



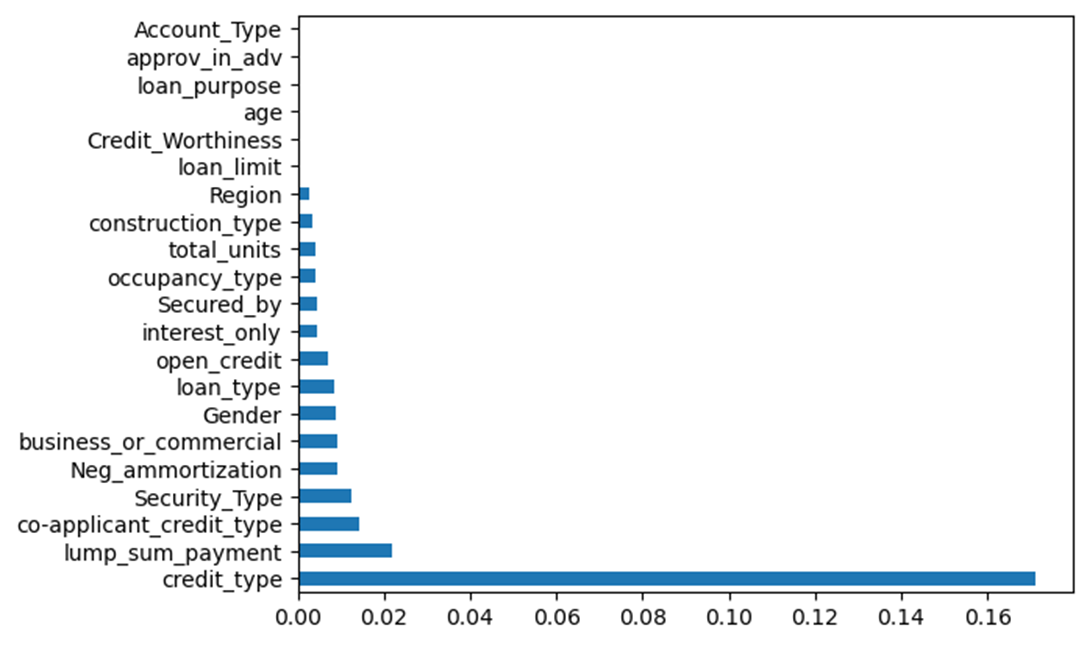
**5.6.3 INFORMATION GAIN**

Information Gain, or IG for short, measures the reduction in entropy or surprise by splitting a dataset according to a given value of a random variable. A larger information gain suggests a lower entropy group or groups of samples, and hence less surprise. Information gain calculates the reduction in entropy or surprise from transforming a dataset in some way.

It is commonly used in the construction of decision trees from a training dataset, by evaluating the information gain for each variable, and selecting the variable that maximizes the information gain, which in turn minimizes the entropy and best splits the dataset into groups for effective classification.

Information gain can also be used for feature selection, by evaluating the gain of each variable in the context of the target variable. In this slightly different usage, the calculation is referred to as mutual information between the two random variables.

We ran these algorithm in our data and the graph below shows the graph of feature importance.Here Credit type is found to have the highest score or higher importance as a deciding factor of the loan default status as compared to other features.

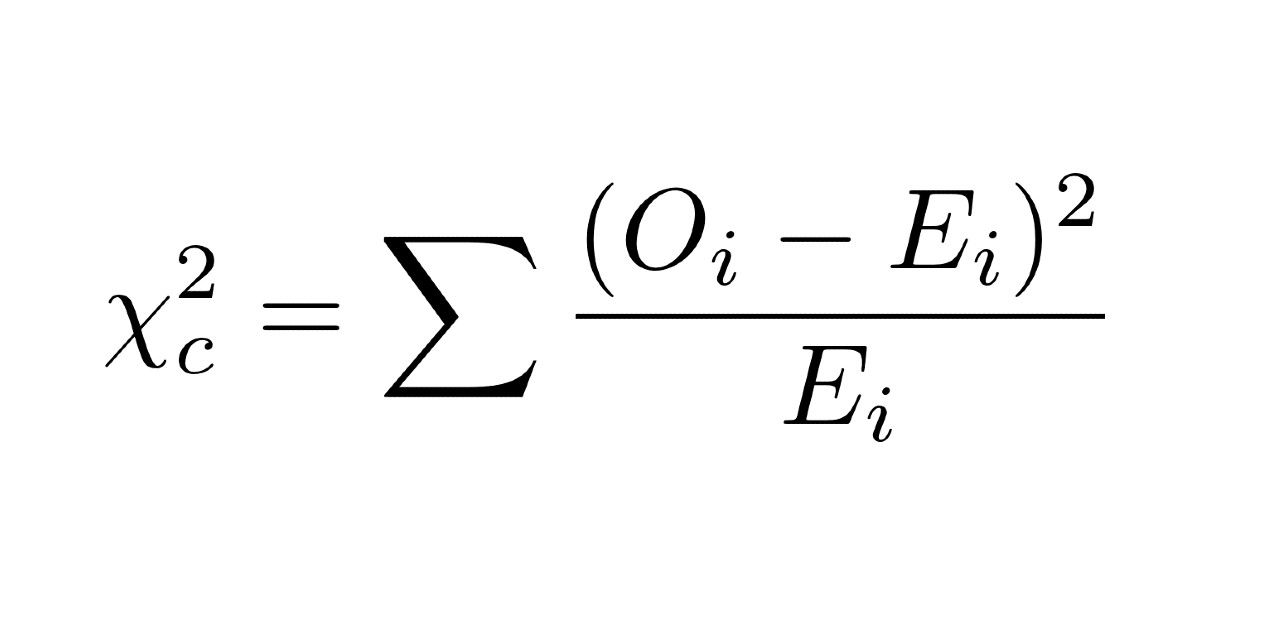


**5.6.4 CHI -SQUARE TEST FOR INDEPENDENCE**

The chi-square test is applied when you have two categorical variables from a single population and evaluates whether there is a significant association between the categories of the two variables.

The chi-square test of independence is used to analyze the frequency table (i.e. contingency table).

Formed by two categorical variables. Chi-square test examines whether rows and columns of a contingency table are statistically associated.

* + **Null hypothesis (H0):** There is no association between the two variables. That means the row and the column variables of the contingency table are independent.
  + **Alternative hypothesis (H1):** there is an association between the two variables. That means the row and column variables are dependent. For each cell of the table, we have to calculate the expected value under the null hypothesis.
  + The formula for the chi-square statistic used in the chi-square test is:
  + 
  + Where “O” is your observed value and “E” is your expected value. It‟s very rare that you‟ll want to actually use this formula to find a critical chi-square value by hand. The summation symbol means that you‟ll have to perform the calculation for every single data item in your data set.
  + We used the Chi-Square test of independence to test the feature relevance under the hypothesis
  + H0:Default Status and the feature is independent
  + H1:Default Status and the features are dependent

Here we took an individual chi square test for each independent feature with the dependent variable “Default Status”. The table below shows the resulting p value from the test.We used a significance level of 5 % to test the hypothesis.Only for Open Credit and Loan Purpose we got a p value greater than the significance level.All other features are found to be significant.That is all other features are dependent on the target column “Default Status”.

| Default Status | p-value |
| --- | --- |
| Loan Limit | <0.01 |
| Gender | <0.01 |
| Approved in advance | <0.01 |
| Loan purpose | 0.106 |
| Loan type | <0.01 |
| Open credit | 0.085 |
| Business or not | <0.01 |
| Credit Worthiness | <0.01 |

**5.6.5 T-TEST OF INDEPENDENT SAMPLES**

The independent t-test, also called the two sample t-test, independent-samples t-test or student's t-test, is an inferential statistical test that determines whether there is a statistically significant difference between the means in two unrelated groups.

## Null and alternative hypotheses for the independent t-test

The null hypothesis for the independent t-test is that the population means from the two unrelated groups are equal:

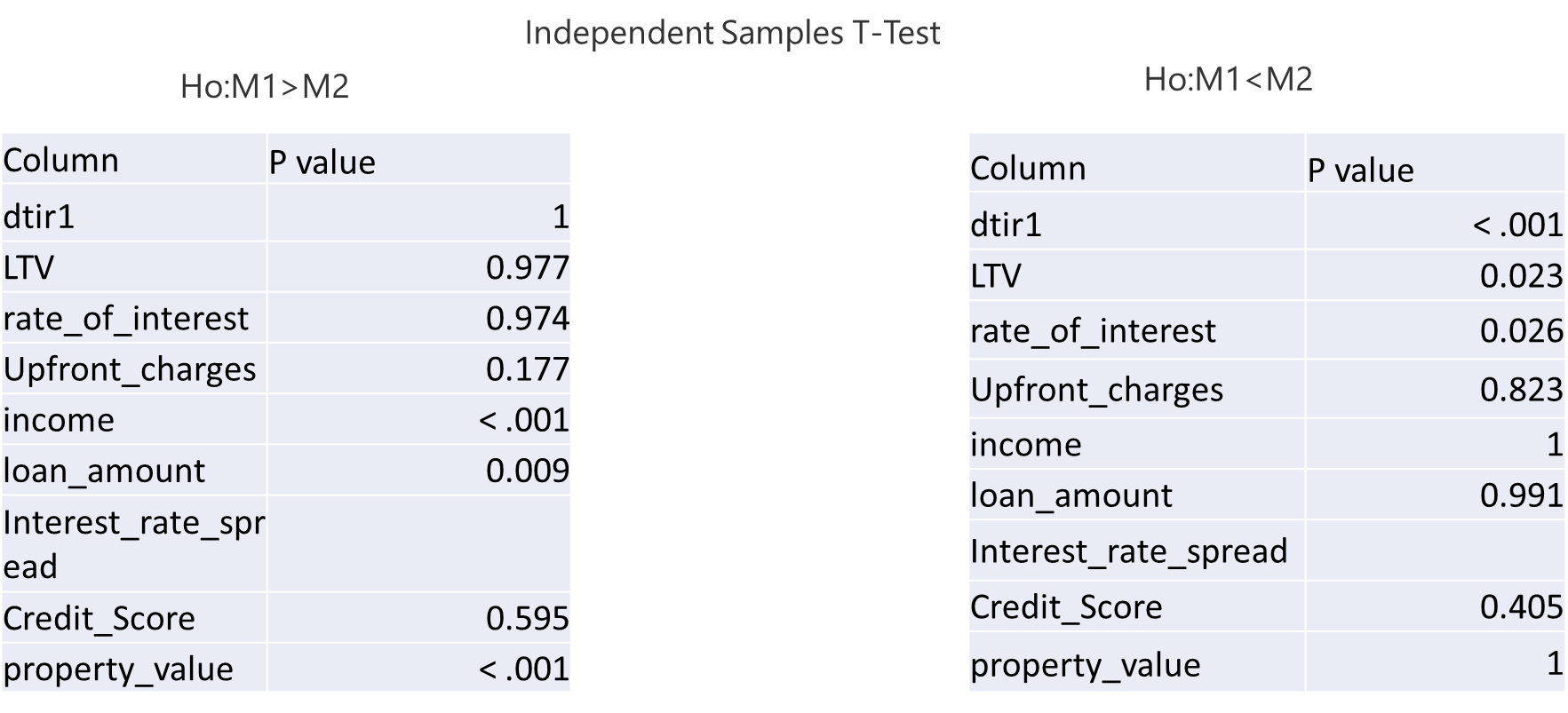
H0: u1 = u2

In most cases, we are looking to see if we can show that we can reject the null hypothesis and accept the alternative hypothesis, which is that the population means are not equal:

HA: u1 ≠ u2

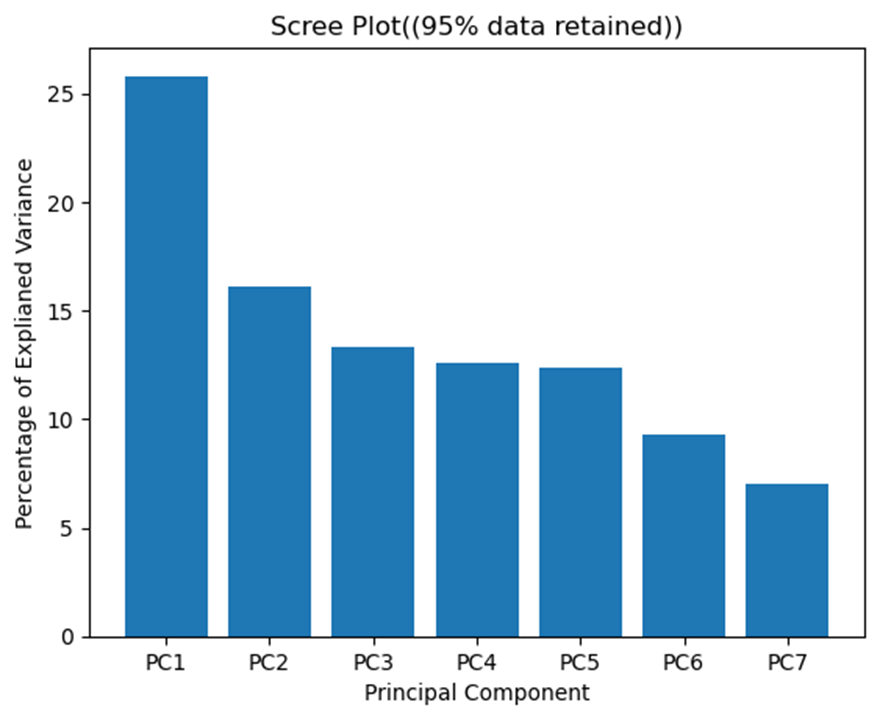
To do this, we need to set a significance level (also called alpha) that allows us to either reject or accept the alternative hypothesis. Most commonly, this value is set at 0.05.

In our data we conducted a T- test for 2 independent samples for finding out whether the features are identically distributed over the Loan-default Status.Here we did this test for the numerical independent columns across the dependent variable.We conducted both one tailed test separately with a significance level of 5%.And from the tables below we could infer that for a right-tail test only income and Credit score had a significant result.And for a left-tail test debt to income ratio,LTV,rate of interest was found to yield a significant result.

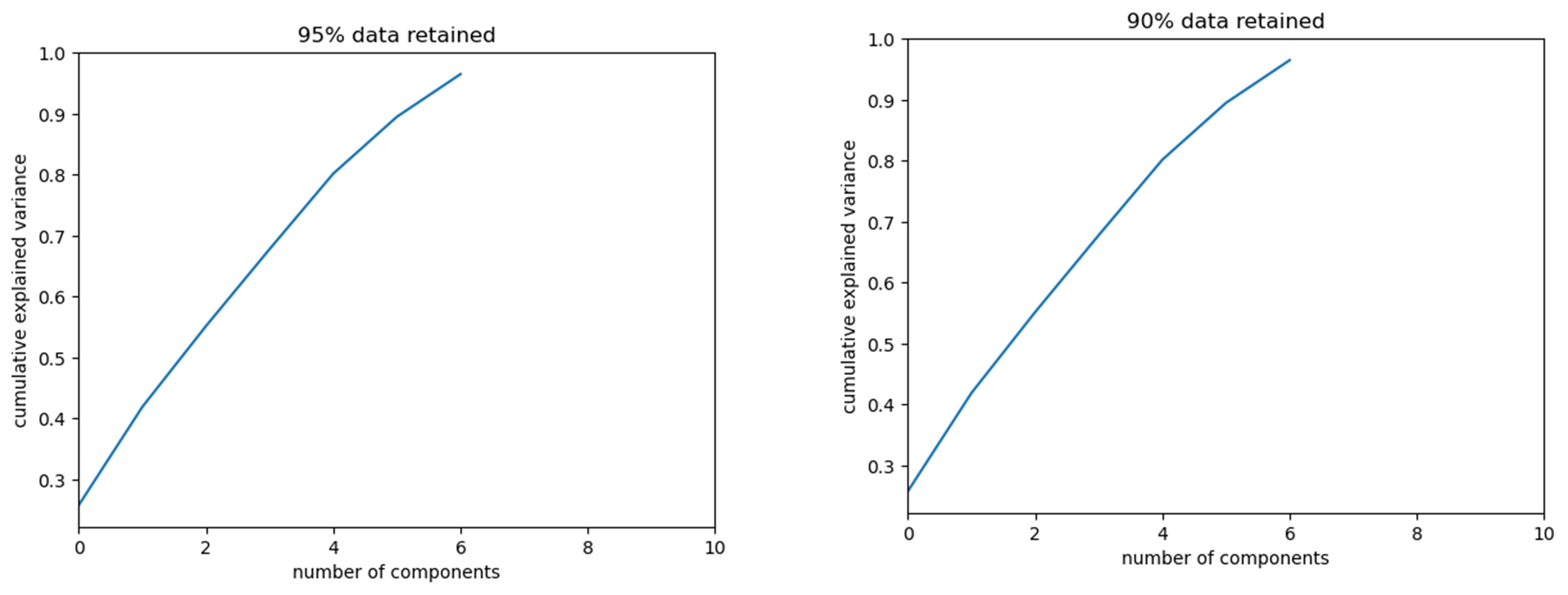


**5.7 DIMENSIONALITY REDUCTION:**

The number of input features, variables, or columns present in a given dataset is known as dimensionality, and the process to reduce these features is called dimensionality reduction. We have done PCA for dimensionality reduction because it increases interpretability yet, at the same time, it minimizes information loss. After doing the PCA at 95% accuracy the shape of data became (10000,24) from (10000,64).



The scree plot for the PCA are shown in the graph below.When 95% data is retained the plot shows that about 7 to 8 principle components will be suffice to explain the whole variations in a data.It is pretty much similar when 90% data is retained.if we take two components then it will be able to explain about 50% of the variations in the data.



**Splitting The data into train and test**

**6. MODEL BUILDING**

A machine learning model could be built by supervised learning algorithm or an unsupervised learning algorithm.In Supervised-Learning data is already tagged with the correct answer. After that, the machine is provided with a new set of examples(data) so that the supervised learning algorithm analyzes the training data(set of training examples) and produces a correct outcome from labeled data.Unsupervised learning is the training of a machine using information that is neither classified or labeled and allowing the algorithm to act on that information without guidance.

Here since the target column “Status”, is present in our dataset,we use a supervised learning algorithm.In turn in a Supervised learning ,there are classification and Regression algorithms .Classification tries to find the decision boundary, which divides the dataset into different classes. Regression algorithms solve regression problems such as house price predictions and weather predictions.Since our target column “Status” is a categorical variable we used classifcation algorithm.

We employed a number of algorithms in our data to predict the target column “Status” of our dataset.

* LOGISTIC REGRESSION
* EXTREME-GRADIENT BOOSTING
* RANDOM FOREST
* DECISION TREE
* SVM
* KNN

**6.1 LOGISTIC REGRESSION**

Logistic regression is a statistical analysis method to predict a binary outcome, such as yes or no, based on prior observations of a data set. A logistic regression model predicts a dependent data variable by analyzing the relationship between one or more existing independent variables.

Logistic regression is used to describe data and explain the relationship between one dependent binary variable and one or more independent variables. Binary logistic regression has three major assumptions:

i. The dependent variable must be either binary or dichotomous (e.g. yes or no).

ii. The data should be free of outliers, which can be determined by converting continuous predictors to standardized scores.

iii. The predictors should not have a lot of strong correlations (multicollinearity). The independent variables must be unrelated to one another. A correlation matrix among the predictors can be used to examine this.

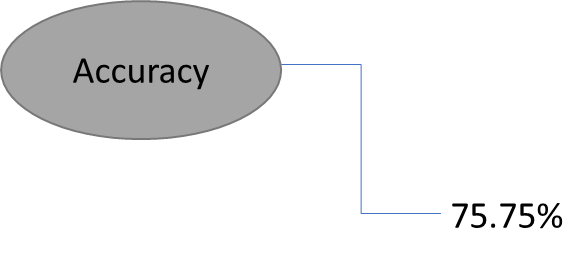
The model is evaluated for across accuracy , precision , recall and F1 score. The model obtained a 100% result in accuracy , 76% result in recall and precision and 86% in F1 Score.

- Accuracy is the total number of correct predictions made to determine if a loan would be default or non-default from the entire sample.

- Recall shows the percentage of non-default predictions are correct over the total number of samples that should have been non-default.

- F1 Score is the harmonic mean between precision and recall.

We did Logistic regression on our data using all the columns,except “ID”, “Year”, “Upfront\_Charges”, “Rate of Spread.And we got an accuracy of 75.74%.



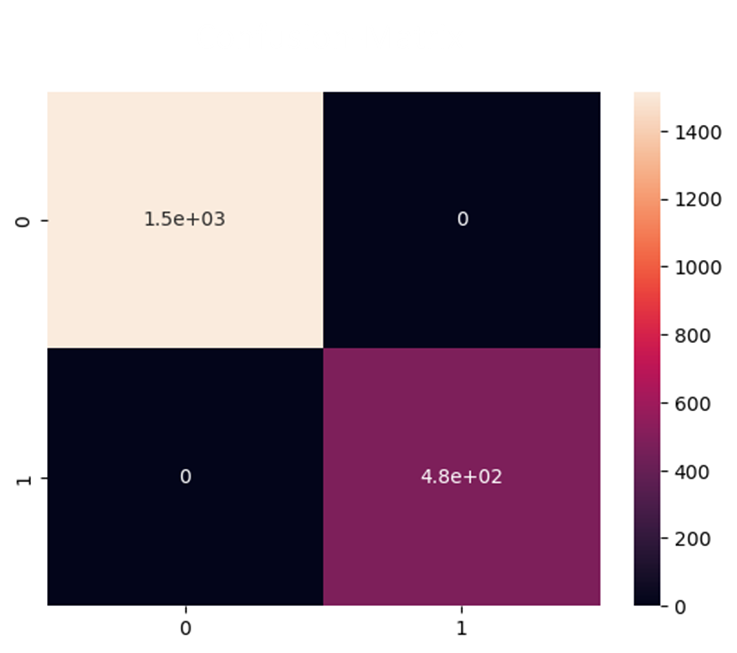
The precision,recall and F1 Score for the logistic regression model is given in the classification report below.

| precision recall f1-score support |
| --- |
| 0 1.00 0.84 0.91 1805 |
| 1 0.40 1.00 0.57 195 |
| accuracy 0.85 2000 |
| macro avg 0.70 0.92 0.74 2000 |
| weighted avg 0.94 0.85 0.88 2000 |

**6.2 EXTREME-GRADIENT BOOSTING**

Ensembles are constructed from decision tree models. Trees are added one at a time to the ensemble and fit to correct the prediction errors made by prior models. This is a type of ensemble machine learning model referred to as boosting.

We employed this model in our data to predict our target variable “Loan Status Default”.This model gives an accuracy of 100%,which is way better than the logistic regression.The confusion matrix showing the TP,FP,TN and FN are shown below.



The classification report given below shows the accuracy,F1 score and other measures that are used to weight a model.



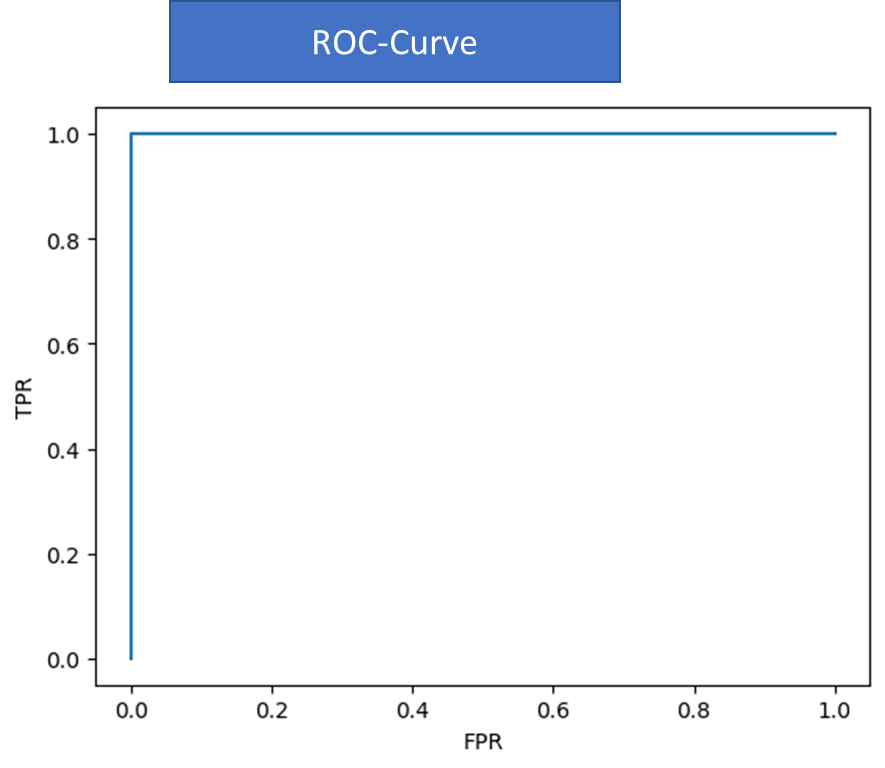
## ROC curve

An ROC curve (receiver operating characteristic curve) is a graph showing the performance of a classification model at all classification thresholds. This curve plots two parameters:

* True Positive Rate
* False Positive Rate

The ROC curve shows the trade-off between sensitivity (or TPR) and specificity (1 – FPR). Classifiers that give curves closer to the top-left corner indicate a better performance.

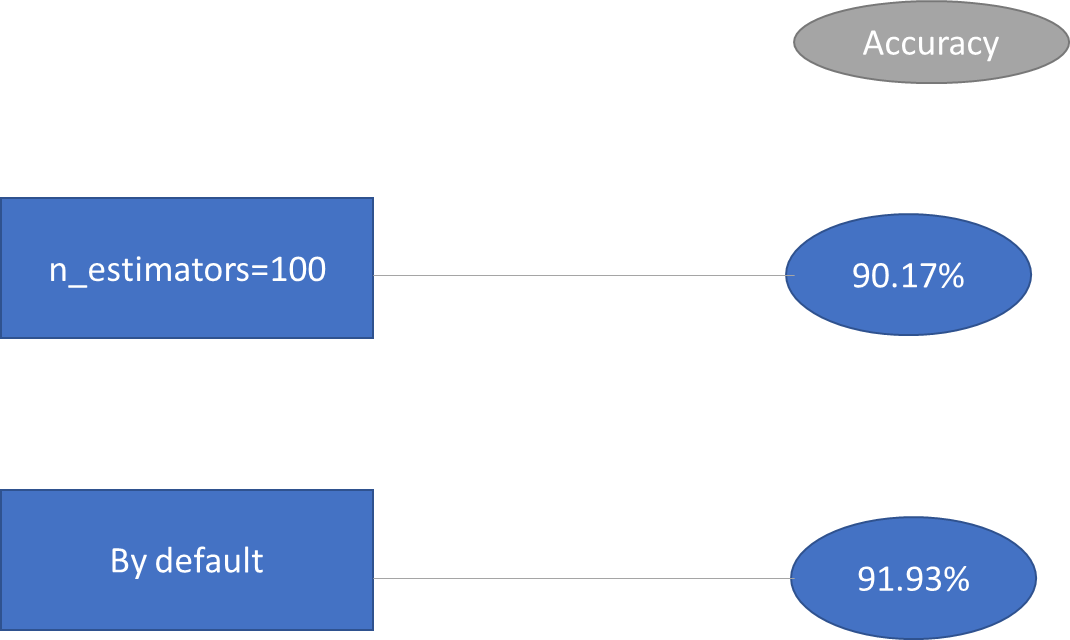
The ROC curve of the EXtreme Gradient Booster is given below. The graph is aligned to the top-left corner indicating a better performed model.



**6.3 RANDOM FOREST**

Random Forest is a supervised machine learning algorithm which is based on ensemble learning. It creates decision trees on the given data samples, gets predictions from each tree and selects the best solution by means of voting.The expected accuracy increases with the number of decision-trees in the model. It is also a good indicator of feature importance.

So, in our data we employed this model in two ways one without giving any attribute and one by giving the number of estimators as 100.The accuracy observed for both the models are shown below.This model is way better than Logistic regression and is less efficient than XGbooster.



**6.4 DECISION TREE**

Decision trees classify instances by sorting them down the tree from the root to some leaf node, which provides the classification of the instance. An instance is classified by starting at the root node of the tree, testing the attribute specified by this node, then moving down the tree branch corresponding to the value of the attribute as shown in the above figure. This process is then repeated for the subtree rooted at the new node.

We have used classification tree for categorical value and a regression tree for the numerical value. In classification trees there are mainly two types , ENTROPY and GINI IMPURITY. for both we got the same accuracy of 90%. Then we have tested whether the model is overfitting or underfitting.

**6.5 SVM**

Support Vector Machine(SVM) is a supervised machine learning algorithm used for both classification and regression. Though we say regression problems as well its best suited for classification. The objective of SVM algorithm is to find a hyperplane in an N-dimensional space that distinctly classifies the data points. The dimension of the hyperplane depends upon the number of features. If the number of input features is two, then the hyperplane is just a line. If the number of input features is three, then the hyperplane becomes a 2-D plane. It becomes difficult to imagine when the number of features exceeds three.

We employed two techniques of SVM in our data one is called SVM(linear-Kernel) and the other one RBF Kernel.A Radial Basis Function (RBF) is a radial function where the reference point is not the origin. For example, distance of 3 from point (5,5).So when we employed the algorithm we got an accuracy of about 76% for both type of SVM.

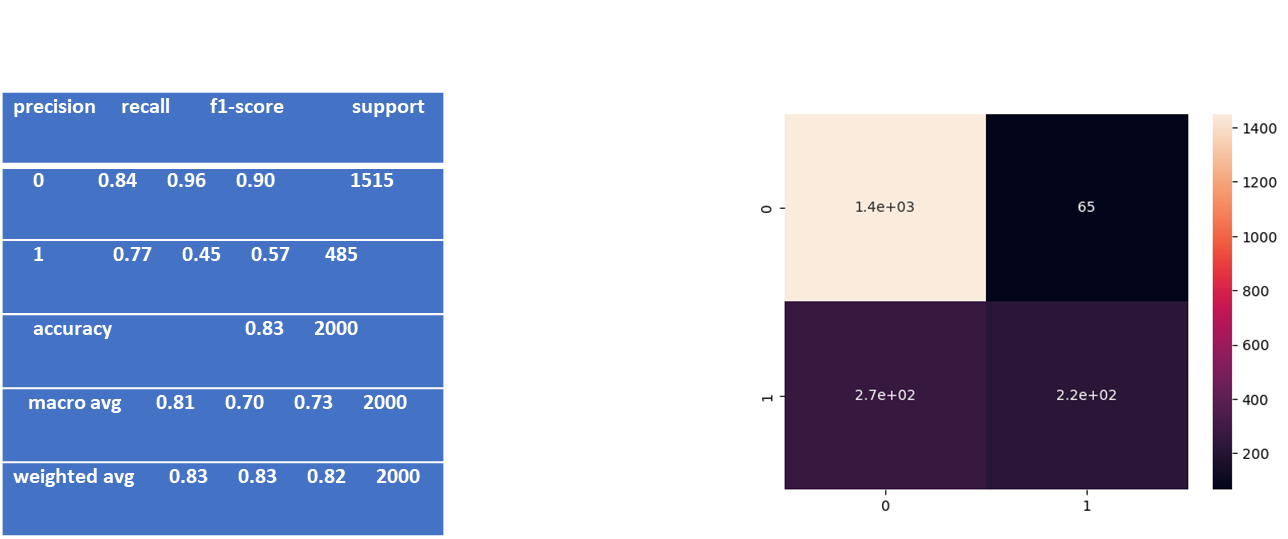
Both of them have an efficiency similar to that of Logistic regression.The classification report is given below.

| **precision recall f1-score support** |
| --- |
| **0 1.00 0.76 0.86 1996** |
| **1 0.00 0.00 0.00 4** |
| **accuracy 0.76 2000** |
| **macro avg 0.50 0.38 0.43 2000** |
| **weighted avg 1.00 0.76 0.86 2000** |

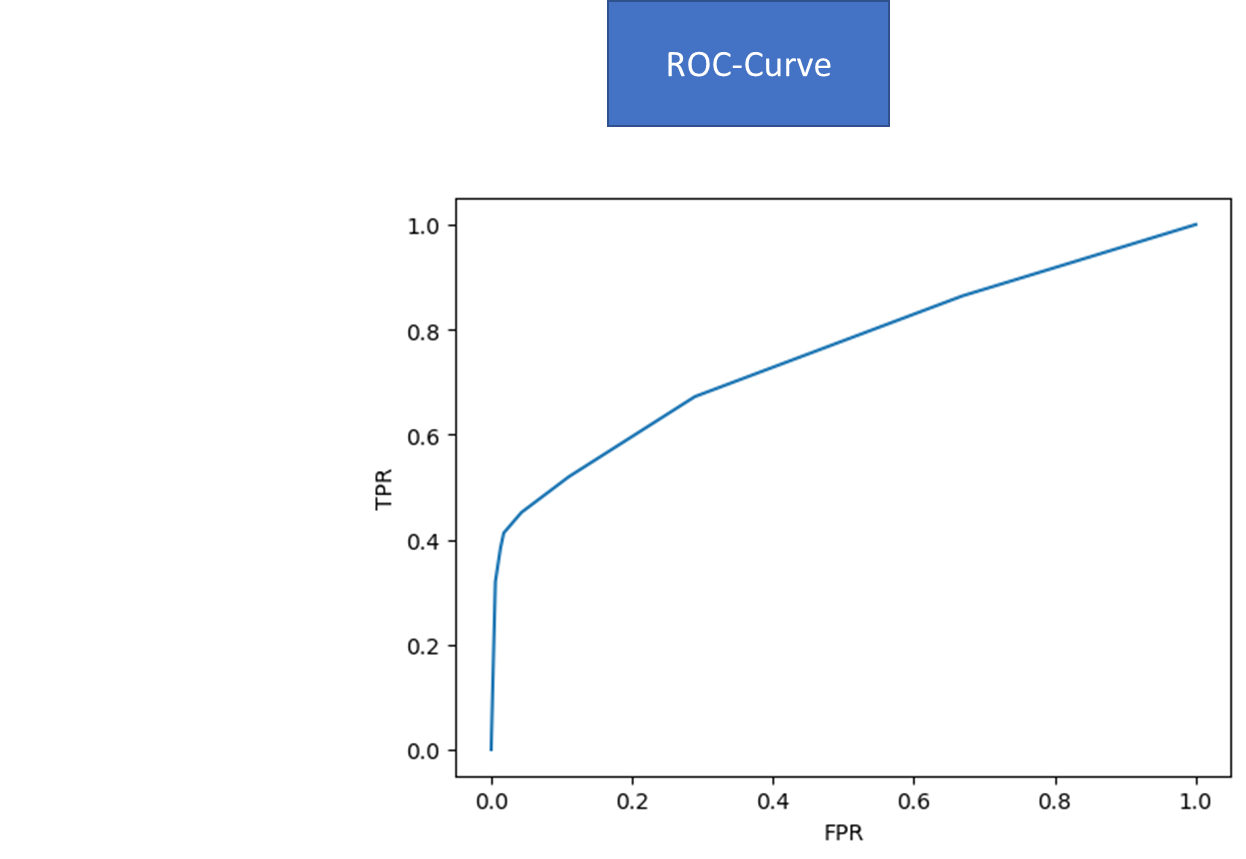
**6.6 KNN**

The K-NN algorithm assumes the similarity between the new case/data and available cases and puts the new case into the category that is most similar to the available categories. K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suited category by using K- NN algorithm.

The Classification report of the KNN-algorithm is shown below.It has got an accuracy of 83%,which is way more efficient than both SVM and Logistic regression.



The confusion matrix of the KNN algorithm containing the TP,FP etc values which help us to evaluate the efficiency of the model is shown above.The ROC-Curve of the KNN-algorithm is given below.



Here from the graph,we could see that the graph is aligning towards the middle indicating a model with a lower efficiency.If the curve has bended towards the top-left it may have been an efficient algorithm to predict our results.

**HYPER PARAMETRIC-TUNING**

A Machine Learning model is defined as a mathematical model with a number of parameters that need to be learned from the data. By training a model with existing data, we are able to fit the model parameters.However, there is another kind of parameter, known as *Hyperparameters*, that cannot be directly learned from the regular training process. They are usually fixed before the actual training process begins. These parameters express important properties of the model such as its complexity or how fast it should learn.

**GridSearchCV**

In the GridSearchCV approach, the machine learning model is evaluated for a range of hyperparameter values. This approach is called GridSearchCV, because it searches for the best set of hyperparameters from a grid of hyperparameters values.

**RandomizedSearchCV**

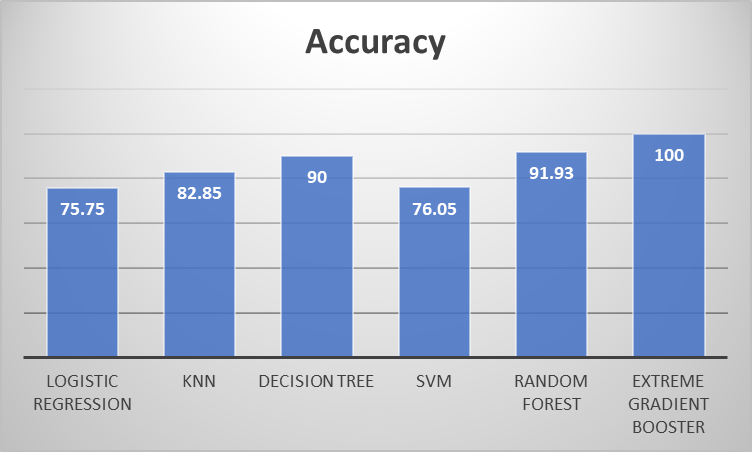
RandomizedSearchCV solves the drawbacks of GridSearchCV, as it goes through only a fixed number of hyperparameter settings. It moves within the grid in a random fashion to find the best set of hyperparameters. This approach reduces unnecessary computation.

We employed both these tuning to our algorithms “Logistic regression” and “Random Forest.

| Model | without tuning | with tuning |
| --- | --- | --- |
| Logistic regression | 75.77% | 86% |
| Random Forest | 91% | 80% |

The accuracy obtained for various Classification models we used is given below .

| **Models** | **Accuracy(%)** |
| --- | --- |
| **Logistic Regression** | **75.75** |
| **K-Nearest Neighbors** | **82.85** |
| **Decision Tree** | **100** |
| **Support Vector Machine (Linear Kernel)** | **75.65** |
| **Support Vector Machine (RBF Kernel)** | **76.05** |
| **Neural Network** | **76.15** |
| **Random Forest** | **100** |
| **Gradient Boosting** | **100** |

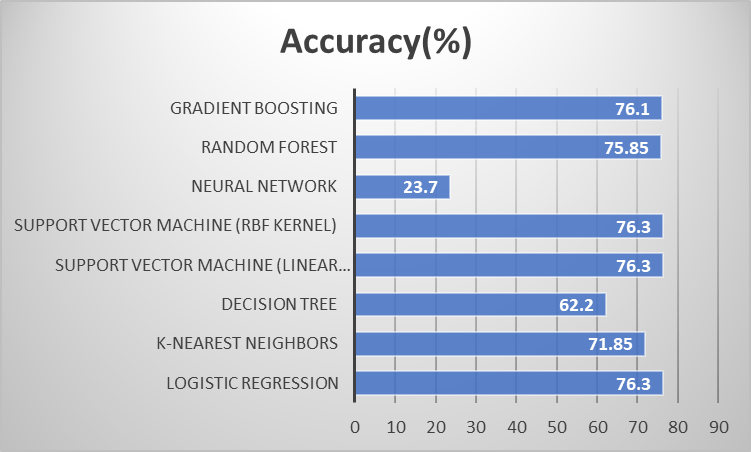
****

Of all the algorithms employed, Extreme Gradient BOOSTER had the highest level of accuracy.So this accuracy score was obtained when we used almost all features for model building.

In order to reduce the dimensionality of the data we used PCA.We employed an algorithm where the dimensionality is reduced keeping 95% of the data is retained.The accuracy we obtained for different models after employing PCA in the data is given below.

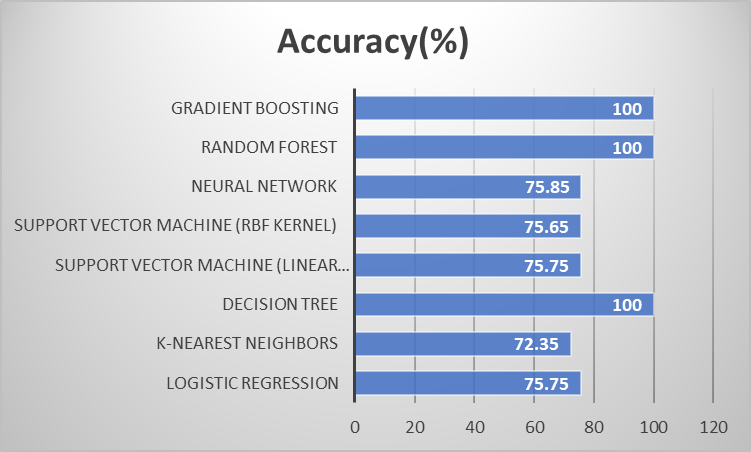
| **Models** | **Accuracy(%)** |
| --- | --- |
| **Logistic Regression** | 76.3 |
| **K-Nearest Neighbors** | 71.85 |
| **Decision Tree** | 62.2 |
| **Support Vector Machine (Linear Kernel)** | 76.3 |
| **Support Vector Machine (RBF Kernel)** | 76.3 |
| **Neural Network** | 23.7 |
| **Random Forest** | 75.85 |
| **Gradient Boosting** | 76.1 |

So,we could observe that the efficiency of the models have considerably reduced when we applied dimensionality reduction technique.But by this way we could rule out the problems of overfitting.XG-Booster,Logistic regression and SVM are performing better than other models in dimensionally reduced data.The accuracy of Neural Network has reduced considerably.



We have also employed a number of pre-Processing techniques to check the feature relevance of the columns..Not all features will be equally relevant in deciding the target column.So by employing feature-relevance techniques like Chi-Square,Information-Gain etc..we find out the most relevant columns for model building.And then performed all the models.The accuracy score for various models by inputting only relevant features is given below.

| **Models** | **Accuracy(%)** |
| --- | --- |
| **Logistic Regression** | 75.75 |
| **K-Nearest Neighbors** | 72.35 |
| **Decision Tree** | 100 |
| **Support Vector Machine (Linear Kernel)** | 75.75 |
| **Support Vector Machine (RBF Kernel)** | 75.65 |
| **Neural Network** | 75.85 |
| **Random Forest** | 100 |
| **Gradient Boosting** | 100 |



We could infer that the models were performing similarly to the initial case where we used all the32 columns.Here Gradient Boosting,Random Forest,Decision tree are have 100% of accuracy, and almost all other models has an accuracy level of 75%.

**7. Result**

1. By inputting PCA performed data,Logistic regression,SVM and XGBooster were found to be the efficient algorithms.

| **Models** | **Accuracy(%)** |
| --- | --- |
| **Logistic Regression** | 76.3 |
| **K-Nearest Neighbors** | 71.85 |
| **Decision Tree** | 62.2 |
| **Support Vector Machine (Linear Kernel)** | 76.3 |
| **Support Vector Machine (RBF Kernel)** | 76.3 |
| **Neural Network** | 23.7 |
| **Random Forest** | 75.85 |
| **Gradient Boosting** | 76.1 |

2)By inputting all the features into the model,decision tree ,Random Forest and XG booster were found to be the better algorithms.

| **Models** | **Accuracy(%)** |
| --- | --- |
| **Logistic Regression** | 75.75 |
| **K-Nearest Neighbors** | 72.35 |
| **Decision Tree** | 100 |
| **Support Vector Machine (Linear Kernel)** | 75.75 |
| **Support Vector Machine (RBF Kernel)** | 75.65 |
| **Neural Network** | 75.85 |
| **Random Forest** | 100 |
| **Gradient Boosting** | 100 |

3)By inputting features having high feature relevance score,Random Forest and XG booster were found to be the better algorithms.

| **Models** | **Accuracy(%)** |
| --- | --- |
| **Logistic Regression** | 75.75 |
| **K-Nearest Neighbors** | 72.35 |
| **Decision Tree** | 100 |
| **Support Vector Machine (Linear Kernel)** | 75.75 |
| **Support Vector Machine (RBF Kernel)** | 75.65 |
| **Neural Network** | 75.85 |
| **Random Forest** | 100 |
| **Gradient Boosting** | 100 |

**WEBSITE LINK**

url: http://127.0.0.1:5000

**8. Conclusion**

* This model was created to check whether the loan is default or not (default or not defaulted) according to many features.
* There were large number of features must be dropped , as:
  + Some of these features were meaningless such as (ID, Year, upfront charges, approved in advance…).
  + Some features had the same meaning and give the same information such as (Credit worthiness & Secured by & property value,occupancy-type)
  + Some features related to the post-process of approval such as (Nag amortization, submission of application…)
* From all of these features (Loan type, loan amount rate of interest, term, property value, income, credit score, age and dirt) were used to check whether the loan is defaulted or not
* We were asked to fill (Nan values) for some features and predict other values for some illogical values (income, interest rate).
* Loan amounts were between (3,576,500 and 16,500) with average interest rate (4.05).
* The largest category applied for alone there ages were between (45-54) for loan of (type 3).
* There is a lightly linear correlation between features and manly strong between (loan amount & property value& income).
* After handling outliers and scaling, we were asked to convert all values to numerical values to be ready to be used for the model.
* Splitting data into train and test considering the scaling process to avoid any bias toward specific data.
* As our target was categorical, we cannot use Linear Regression model.
* TheXG-Booster, Decision tree model and Random forest model gave the best accuracy.
* Our first trial to drop features and depending on (18 feature) , we were suffering from overfitting ( train 98% & test 75% ) and can be handled by Cross validation and gave ( train 100% & test 100% ).
* After reducing the features to and depending on the only affected features and it was a way of reducing overfitting, the accuracy was (train 93% & test 83%).
* By applying Cross validation, we got accuracy of 99% for average training score and 97% for average testing score.
* Alternativelly &Finally , using very simple model without scaling gave a very high accuracy without the needing for applying cross validation, as Decision Trees & Random foreast don't need it actually. Because, they just try to find a threshold value for a given feature that best splits the samples. And, whether you scale it or not, a similar threshold will be chosen, since the ordinality of the variables doesn't change(they do not require feature scaling to be performed as they are not sensitive to the the variance in the data.)

**References**

1. Brownlee, J. (2021, January 5). Random Oversampling and Undersampling for Imbalanced Classification. Machine Learning Mastery. Retrieved from https://machinelearningmastery.com/random-oversampling-and-undersampling-forimbalanced-classification/
2. Camilla Cal`ı and Maria Longobardi. “Some mathematical properties of the ROC curve and their applications”. In: Ricerche di Matematica 64 (Oct. 2015). doi: 10.1007/s11587-015-0246-8.
3. Chakure, A. (2019, June 29th). Random Forest Regression: Along with its implementation in Python. Medium. Retrieved from <https://medium.com/swlh/random-forest-and-itsimplementation-71824ced454f>
4. Gandhi, R. (2018a.) Boosting Algorithms: AdaBoost, Gradient Boosting and XGBoost. Medium. Retrieved from

https://medium.com/hackernoon/boosting-algorithms-adaboost-gradient-boosting-and-xgboost-f74991cad38c

1. Kjell Kuhn Max Johnson. Feature Engineering and Selection: A Practical Approach for Predictive Models. eng. Feature Engineering and Selection: A Practical Approach for Predictive Models. 2019. isbn: 9781138079229.
2. Larose, D.T. & Larose, C.D. (2015). Data mining and predictive analytics. John Wiley & Sons, Incorporated
3. L. C. Thomas, E. N. C. Tong, and C. Mues. “Mixture cure models in credit scoring: If and when borrowers default”. In: European Journal of Operational Research 218 (2012), pp. 132–139.
4. Michael Bowles. Machine Learning in Python: Essential Techniques for Predictive Analysis. eng. 2015. isbn: 9781118961742.
5. Tom M Mitchell. “Machine learning and data mining”. In: Communications of the ACM 42.11 (1999).
6. <http://scikit-learn.org/stable/modules/classes.html#module-sklearn.metrics>
7. <https://machinelearningmastery.com/logistic-regression-for-machine-learning/>
8. <http://scikit-learn.org/stable/modules/generated/sklearn.model_selection.KFold.html>
9. <http://scikit-learn.org/stable/modules/generated/sklearn.model_selection.cross_val_predict.html>